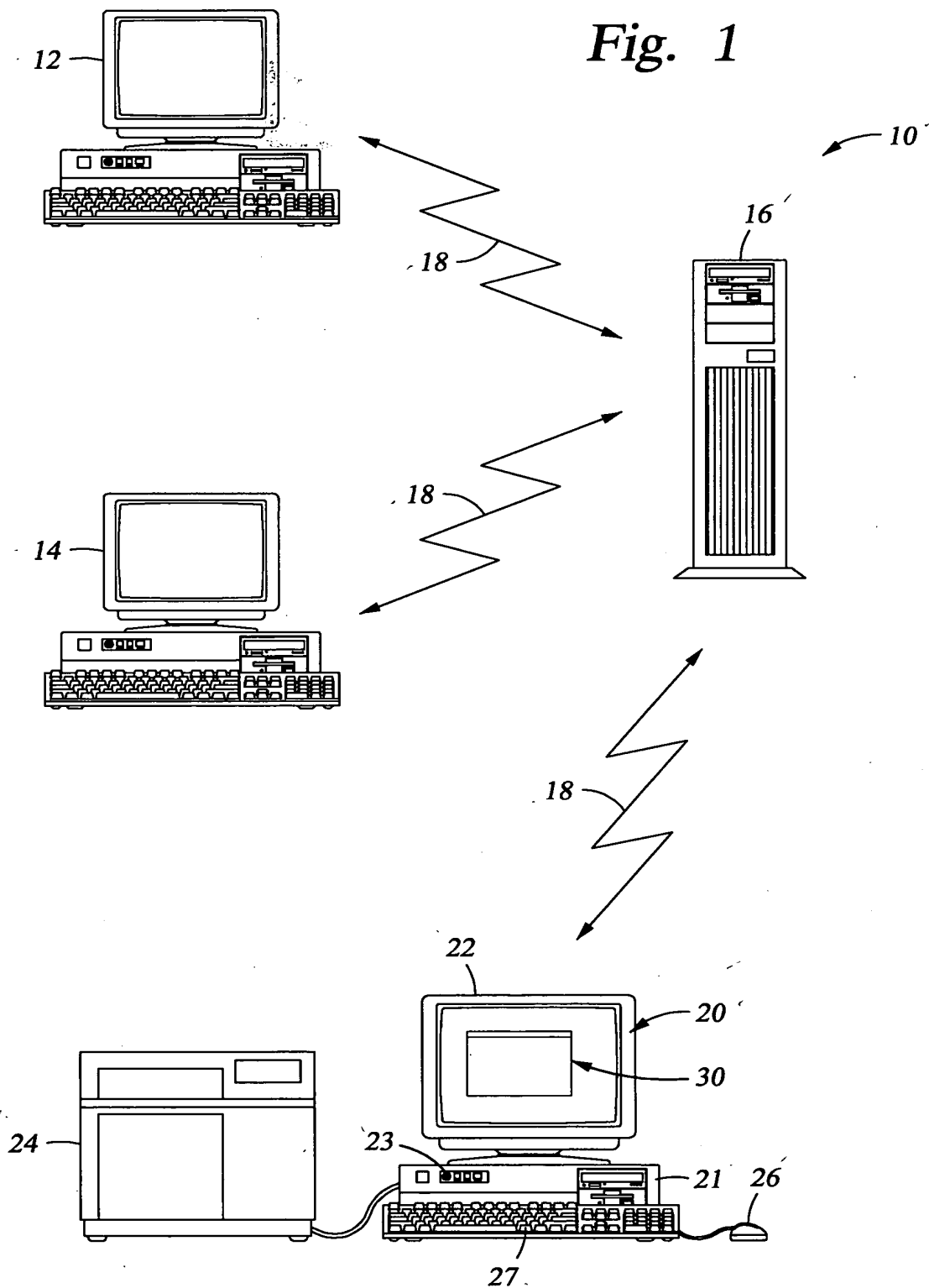


Fig. 1



005250" 0205/050

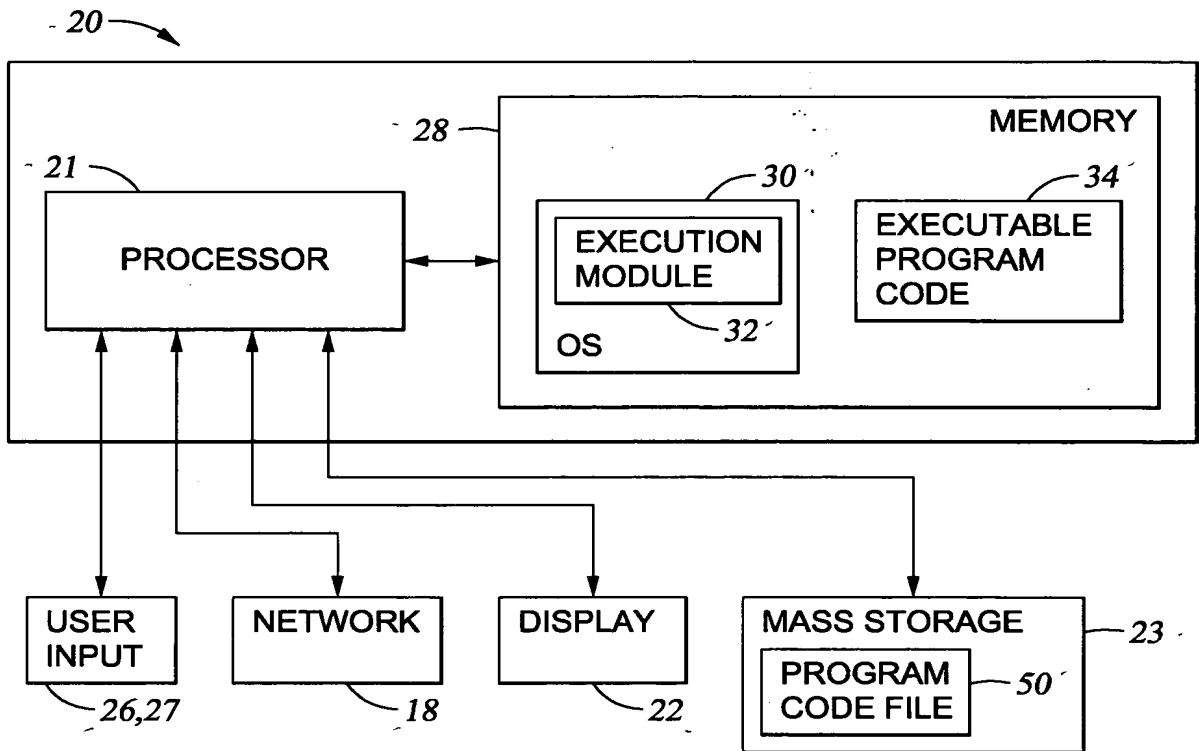


Fig. 2

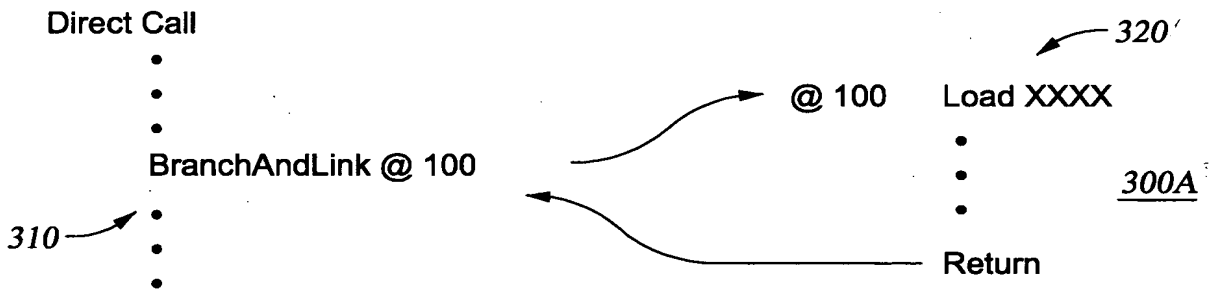


Fig. 3A

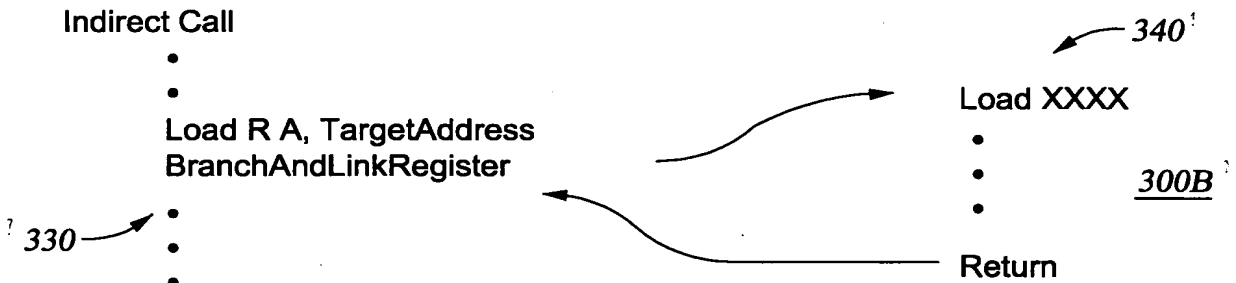


Fig. 3B

Inlining

•

•

•

Load R1,X

Load R2, Y

BranchAndLink @ 100

•

•

•

•

•

•

Load R1,X

Load R2, Y

Add R3,R1,R2

•

•

•

→ @100

Add R3,R1,R2

400

Return

Fig. 4

Inlining (Continued)

•

•

•

LoadImmediate R1,5

LoadImmediate R2,6

BranchAndLink @ 100

•

•

•

•

•

•

LoadImmediate R3,11

•

•

•

→ @100

Add R3,R1,R2

500

Return

Fig. 5

MULTIPLE COPIES OF STATIC

Fig. 6A

```

Class A {
    static lint X= 5;
    static synchronized int getNext() { 600A
        X++;
        return X;
    }
}

```

Fig. 6B

```

class Aprime {
    static lint X= 5;
    static synchronized int getNext() { 600B
        X++;
        return X;
    }
}

```

Virtual Call

-
-
-

Load RA, offset of MethodRef in constant pool(RS)
 Load RB, offset of VMT offset in MethodRef(RA)

Load RG, Ø (RO)
 Add RD, RB, RC
 Load RE, Ø (RD)

1310

Load RF, offset to entry point ptr in MethodRef(RE)
 Move RP0, RE
 Move RT, RF
 BranchIndirect

Fig. 13

SEPARATELY COMPILED COPIES OF THE SAME CLASS

```

class A {
    static int doSomething() {
        int Y = B.someMethod() - C.anotherMethod();
        .
        .
        .
    }
}

```

Static as compiled for first copy:

```

    .
    .
    .
@ 11 ptr to B.someMethod
@ 12 ptr to C.anotherMethod
    .
    .
    .

```

700

Static as compiled for fsecond copy:

```

    .
    .
    .
@ 11 ptr to C.anotherMethod
@ 12 ptr to B.someMethod
    .
    .
    .

```

Fig. 7

TRADITIONAL COMMON ADDRESS SPACE SCHEME

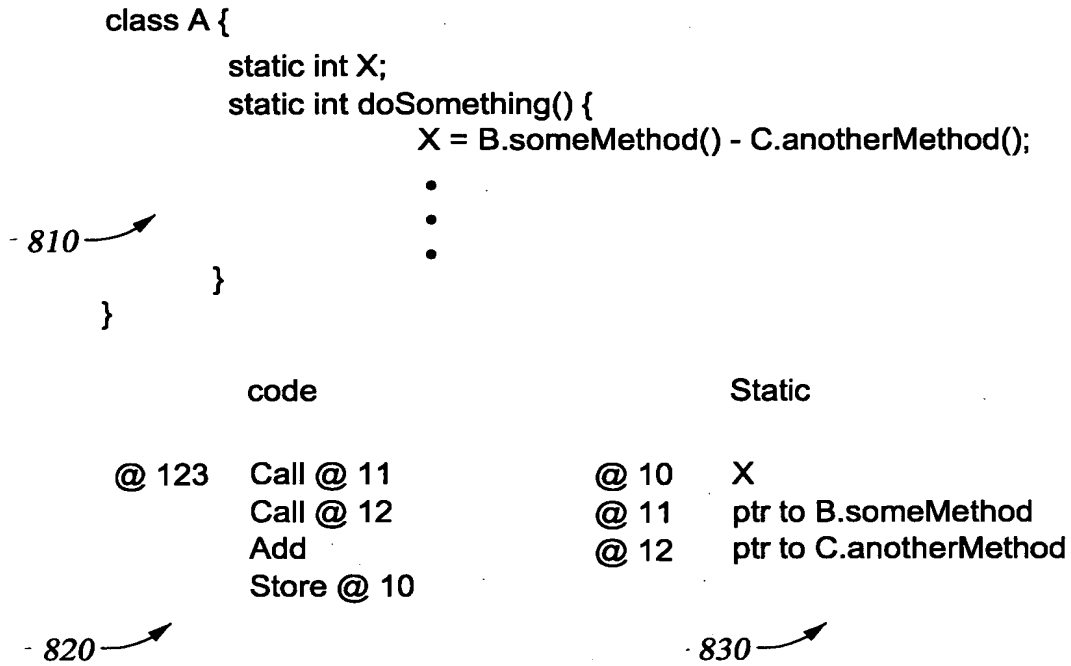


Fig. 8A

TRADITIONAL COMMON REENTRANT IMAGE SCHEME

(Same source)

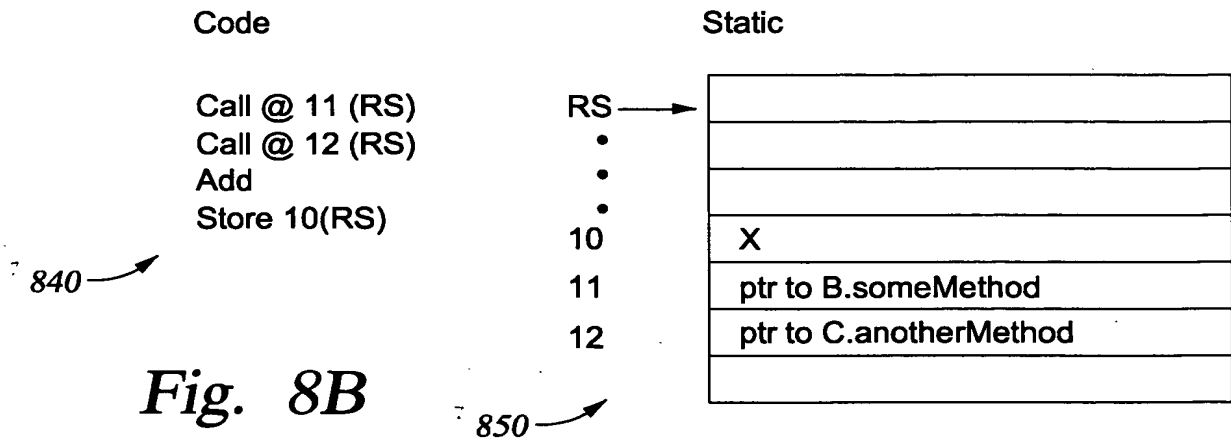
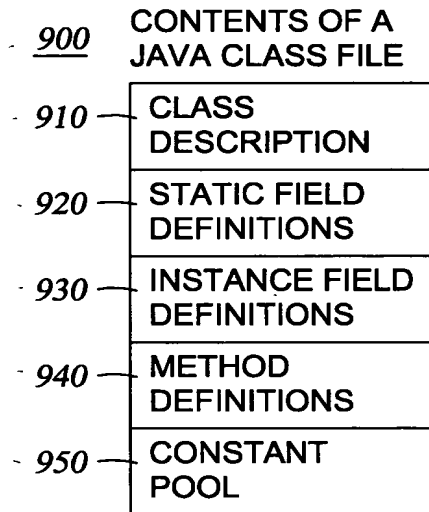
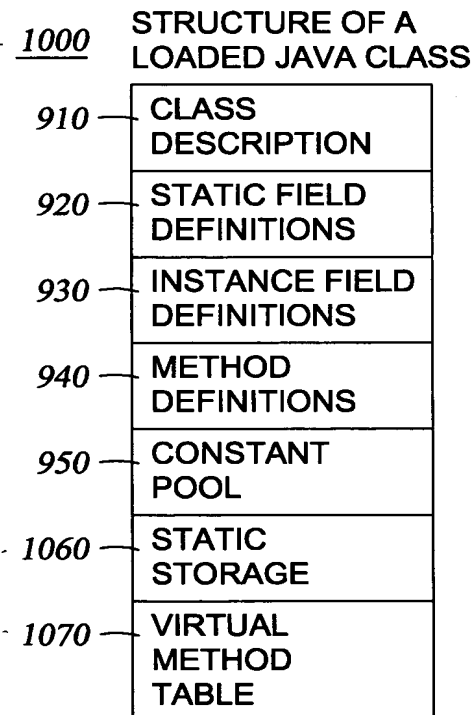
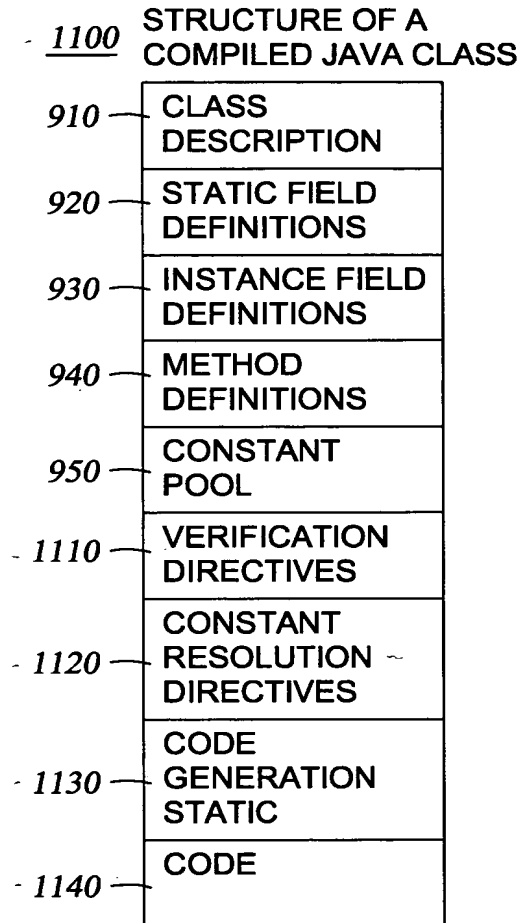


Fig. 8B

*Fig. 9**Fig. 10**Fig. 11*

Indirect Call

-
-
-

Load RA, offset of MethodRef in constant pool(RS)

Load RB, offset of MethodInfo ptr in MethodRef(RA)

Load RC, offset to entry point ptr in MethodInfo(RB)

Move RP0 RB

Move RT, RC

BranchAndLink Indirect

-
-
-

1210

Direct Call

-
-
-

Load RA, offset of MethodRef in constant pool(RS)

Load rb, offset of MethodInfo ptr in MethodRef(RA)

Move RP0, RB

BranchAndLink Direct @ of entry

1220

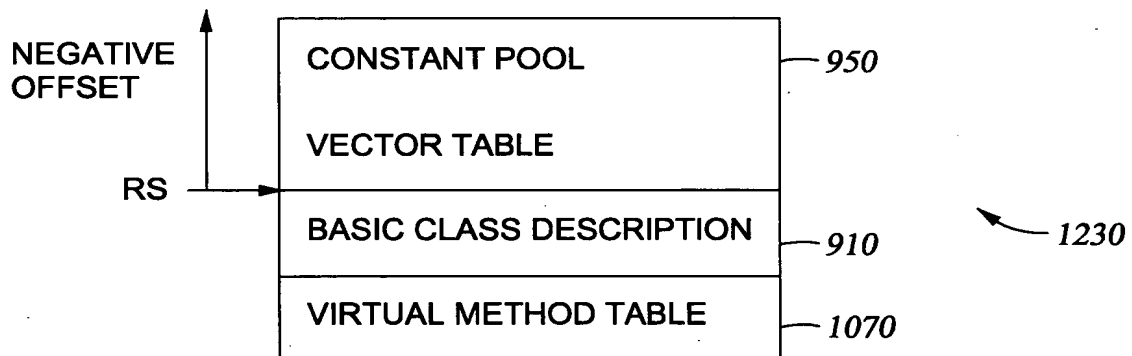


Fig. 12

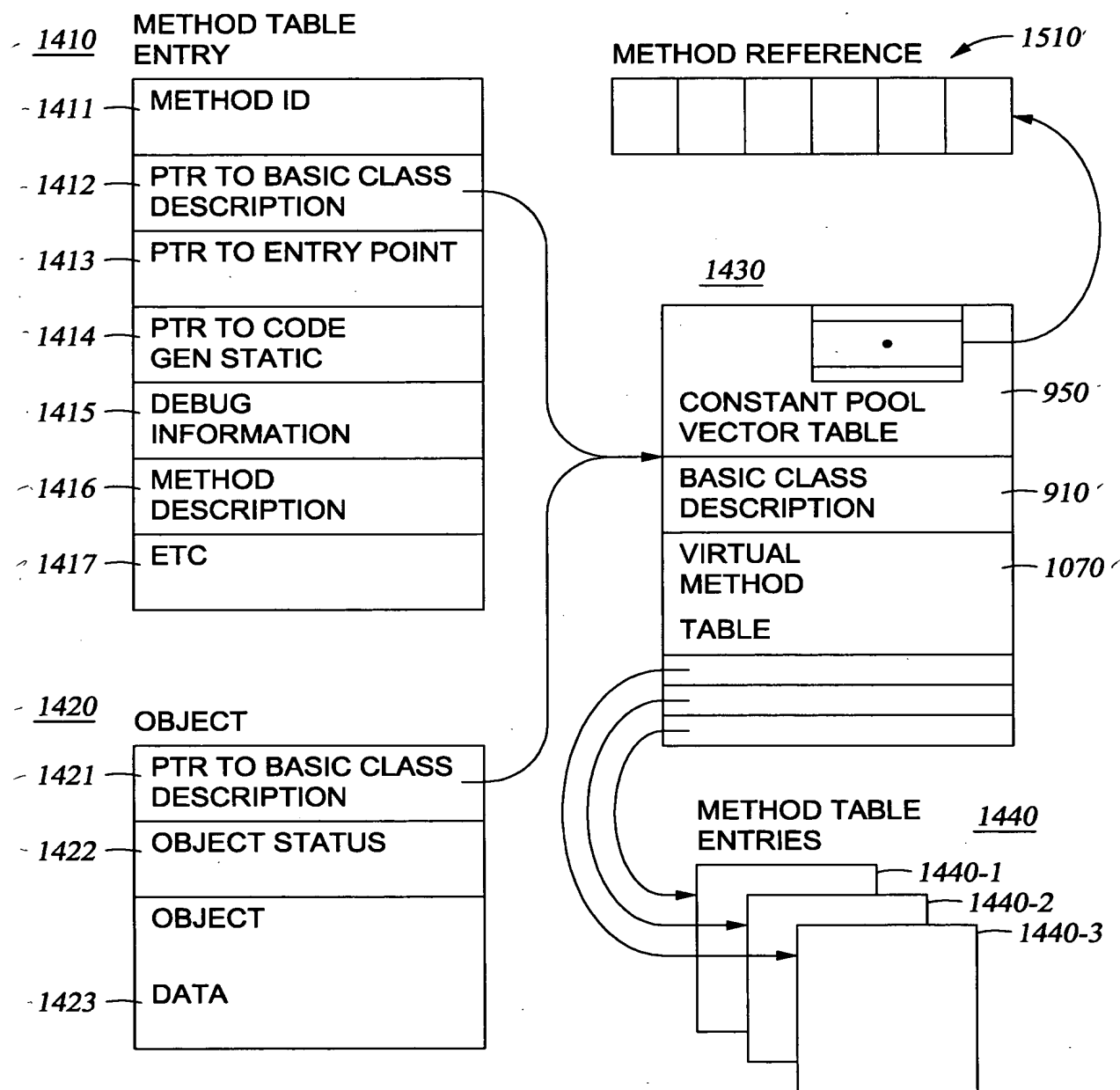


Fig. 14

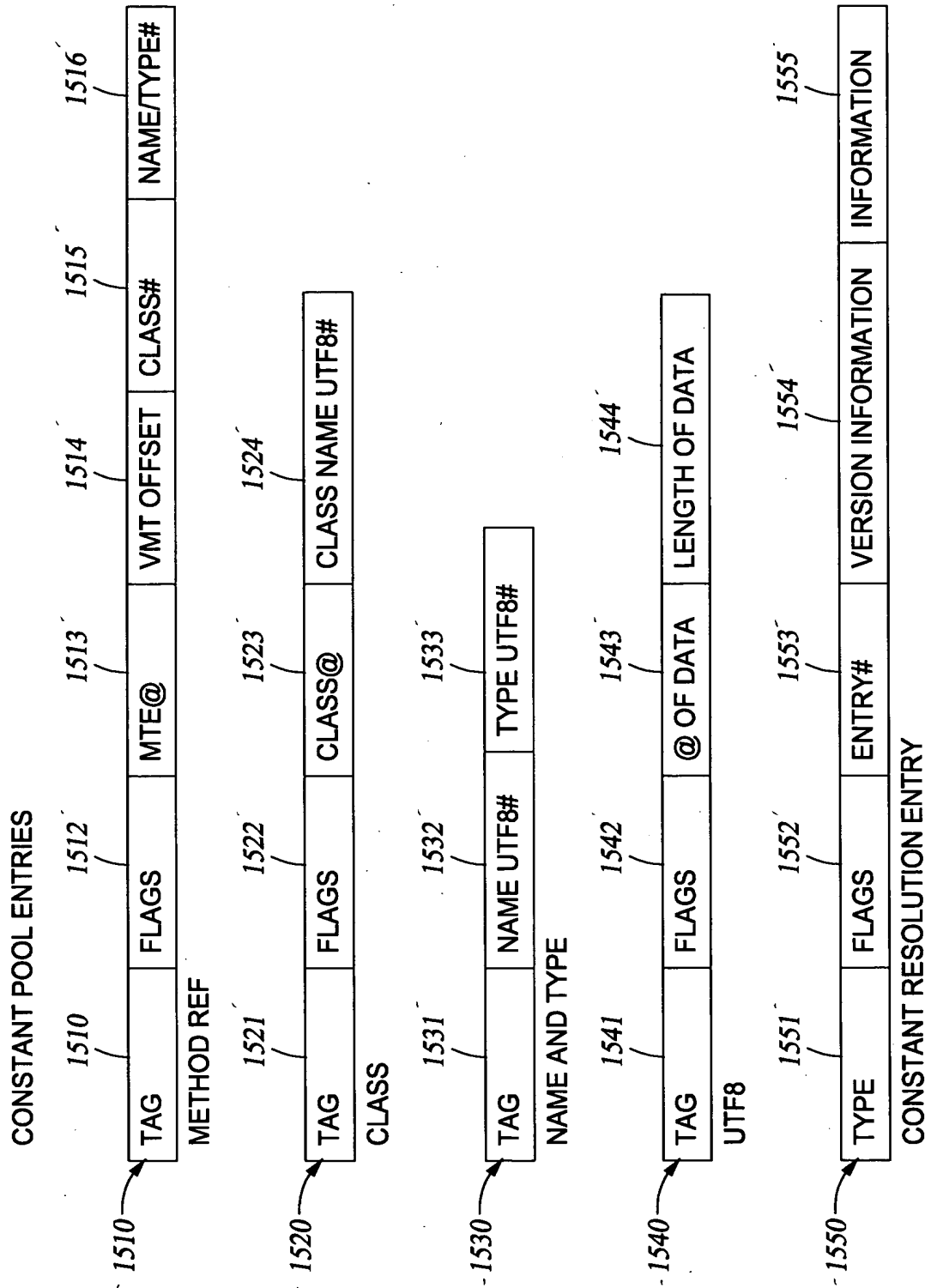


Fig. 15

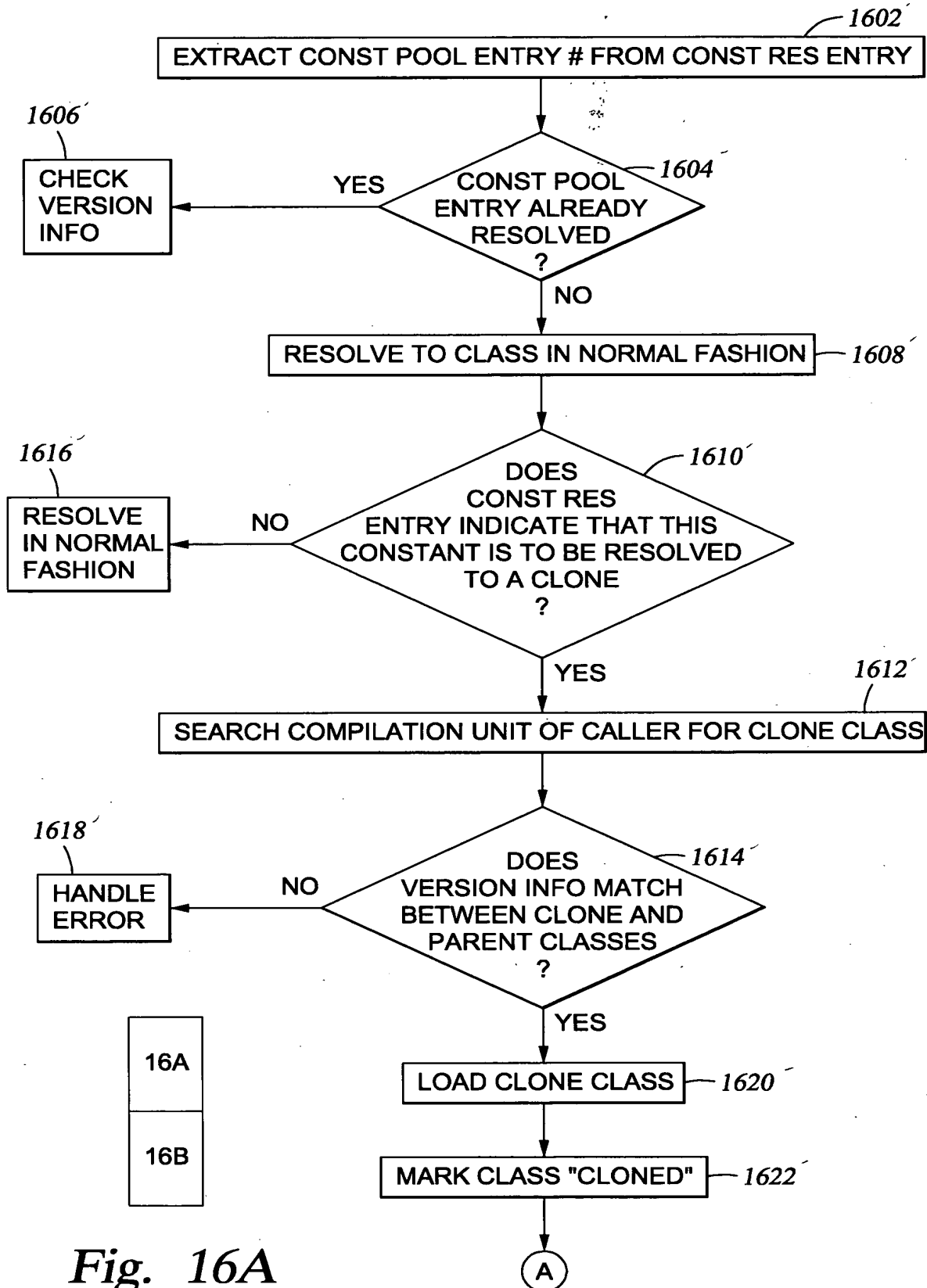
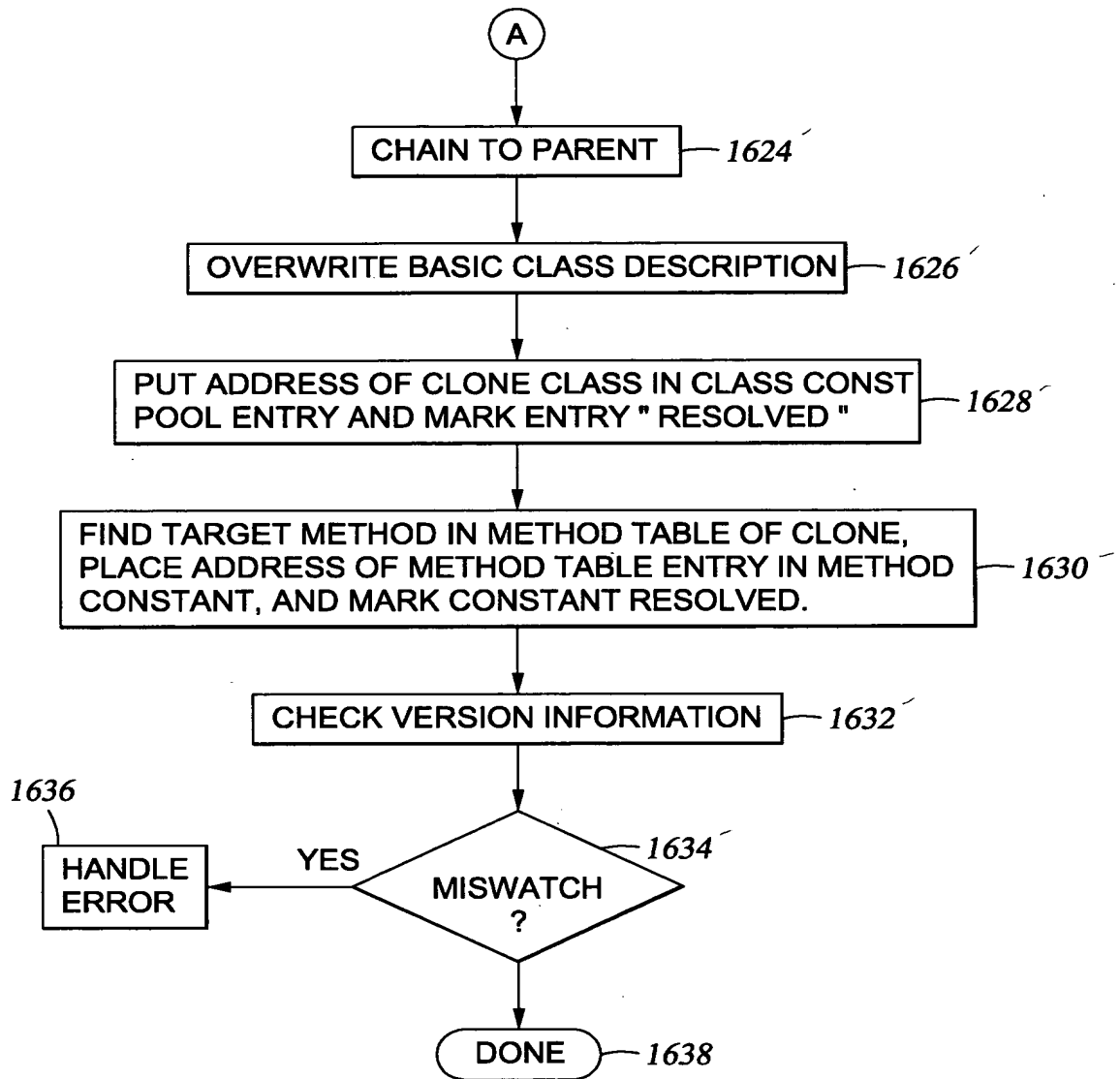


Fig. 16A

*Fig. 16B*

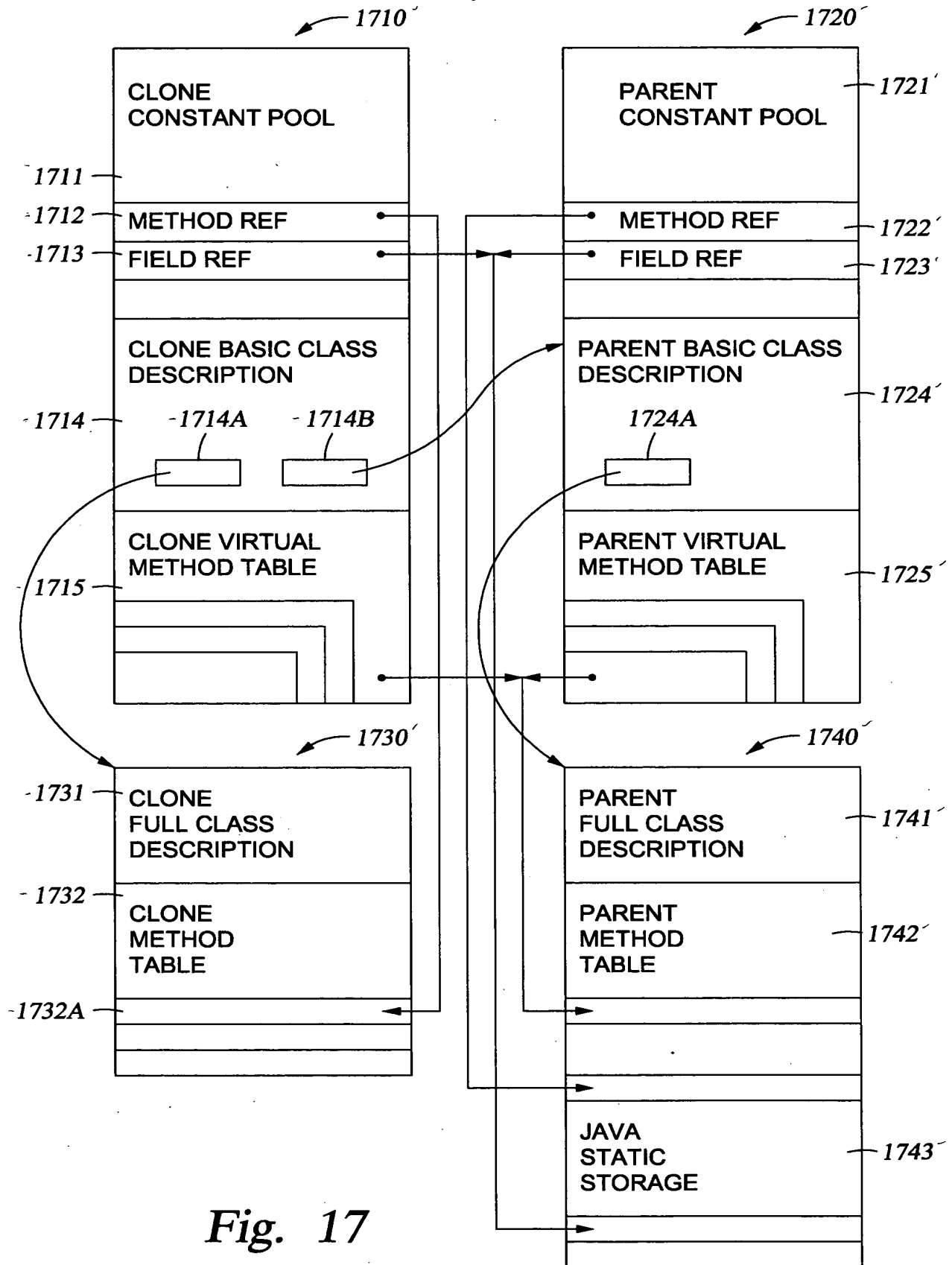
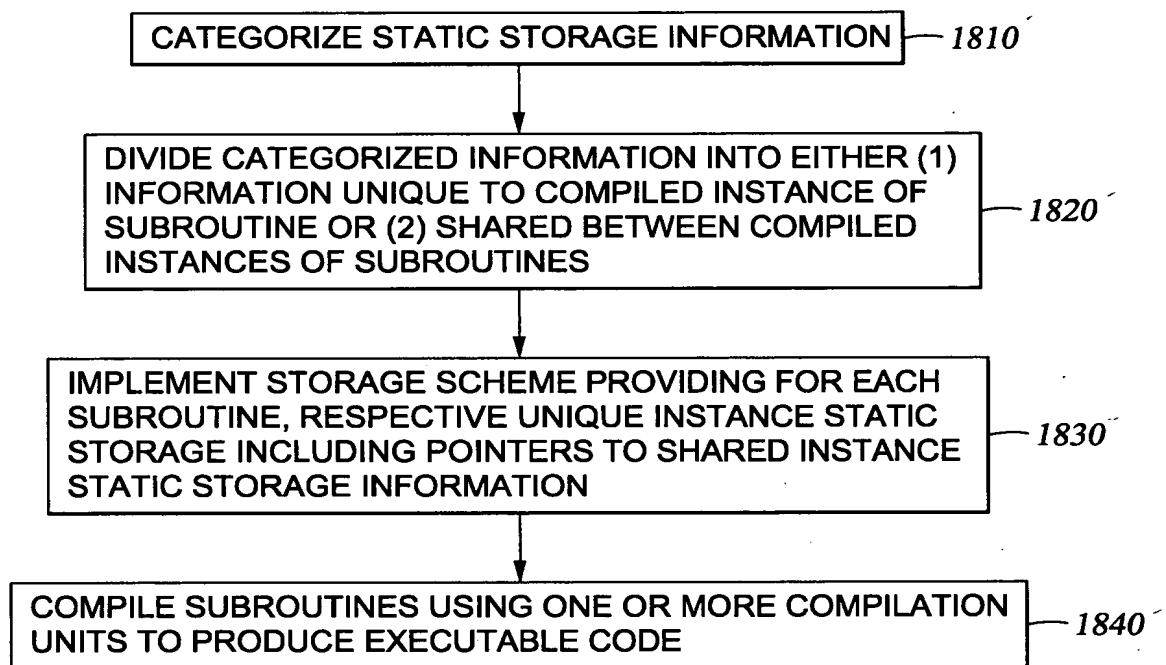
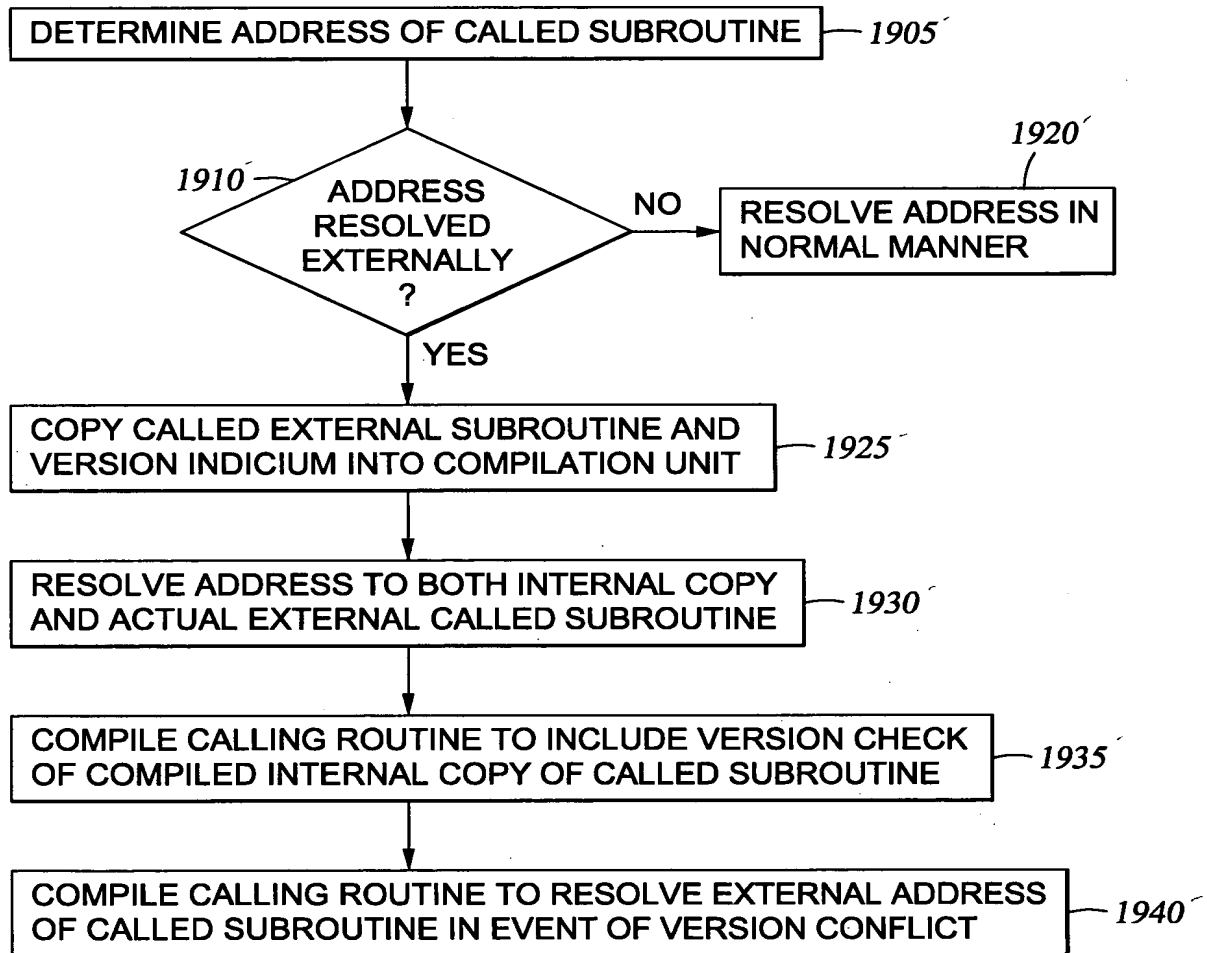


Fig. 17

*Fig. 18*

1900*Fig. 19*